

School Page API

ATTENTION!!!

After the user successfully logs in, the system will automatically issue a JWT token. For every subsequent request, this token needs to be included in the request header sent to the server. The header name should be `token`, and its value should be the JWT token issued at login. If the system detects that the user is not logged in, it will return the following fixed error message:

```
1 {"code":0,"msg":"NOT_LOGIN"}
```

1. Paginated Query for Schools

1.1 Basic Information

Request Path: `/schools`

Request Method: `GET`

API Description: This API is used to paginated query schools.

1.2 Request Parameters

Format: `queryString`

Request Parameter Example:

```
1 /schools?page=1
```

or

```
1 /schools?page=1&name=xxx
```

Note: The `name` parameter is optional. If the `name` parameter is set, a fuzzy search will be performed in the database to find related data. The `page` parameter is compulsory. The page size is fixed(10).

1.3 Response Data

Format: `application/json`

Response Parameter Example:

```
1 {
2   "code": 1,
3   "msg": "success",
4   "data": {
5     "total": 7,
6     "rows": [
7       {
8         "schoolId": 10,
9         "name": "school10",
10        "postCode": "11111",
11        "icsea": "22222",
12        "abnNum": "33333"
13      },
14      {
```

```
15         "schoolId": 11,  
16         "name": "school11",  
17         "postCode": "11111",  
18         "icsea": "22222",  
19         "abnNum": "33333"  
20     }  
21 ]  
22 }  
23 }
```

2. Delete School by ID

2.1 Basic Information

Request Path: /schools/{id}

Request Method: DELETE

API Description: This API is used to delete a school by its id.

2.2 Request Parameters

Format: Path variable

Request Parameter Example:

```
1 /schools/1
```

2.3 Response Data

Format: application/json

Response Parameter Example:

```
1 {  
2     "code": 1,  
3     "msg": "success",  
4     "data": null  
5 }
```

3. Add new School

3.1 Basic Information

Request Path: /schools

Request Method: POST

API Description: This API is used to add a new school.

3.2 Request Parameters

Format: application/json

Request Parameter Example:

```
1 {
```

```
2     "name": "add school",
3     "postCode": "add",
4     "icsea": "add",
5     "abnNum": "add"
6 }
```

3.3 Response Data

Format: application/json

Response Parameter Example:

```
1 {
2     "code": 1,
3     "msg": "success",
4     "data": null
5 }
```

4. Query School by id

4.1 Basic Information

Request Path: /schools/{id}

Request Method: GET

API Description: This API is used to query a school by its id.

4.2 Request Parameters

Format: Path Variable

Request Parameter Example:

```
1 /schools/7
```

4.3 Response Data

Format: application/json

Response Parameter Example:

```
1 {
2     "code": 1,
3     "msg": "success",
4     "data": {
5         "schoolId": 7,
6         "name": "school7",
7         "postCode": "11111",
8         "icsea": "22222",
9         "abnNum": "33333"
10    }
11 }
```

5. Update School

5.1 Basic Information

Request Path: /schools

Request Method: PUT

API Description: This API is used to update school.

5.2 Request Parameters

Format: application/json

Request Parameter Example:

```
1 {  
2   "schoolId": "17",  
3   "postCode": "1111",  
4   "icsea": "1111",  
5   "abnNum": "1111"  
6 }
```

5.3 Response Data

Format: application/json

Response Parameter Example:

```
1 {  
2   "code": 1,  
3   "msg": "success",  
4   "data": null  
5 }
```